Fig.1

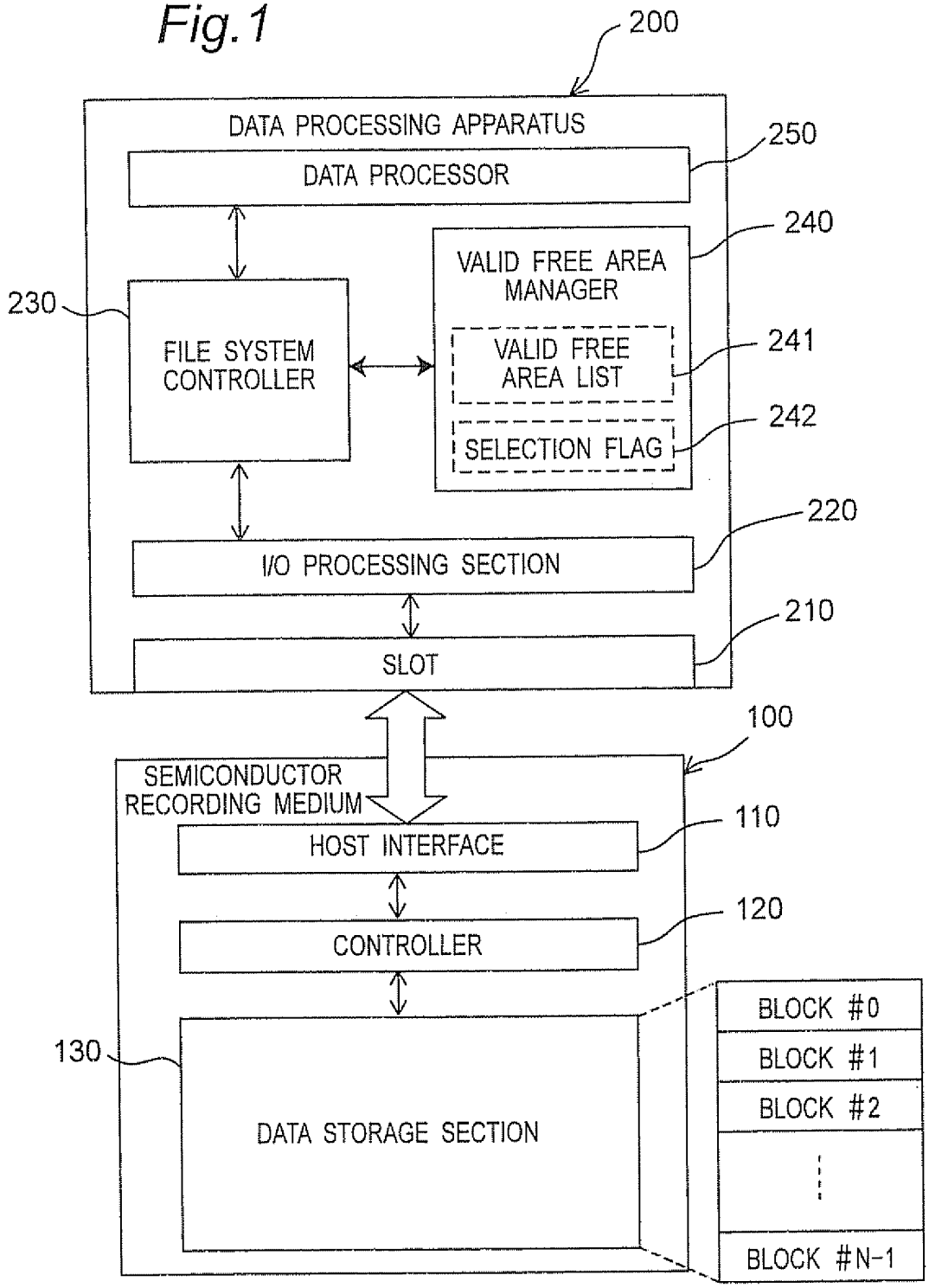
200
DATA PROCESSING APPARATUS
250
DATA PROCESSOR
240
VALID FREE AREA MANAGER
230
FILE SYSTEM CONTROLLER
241
VALID FREE AREA LIST
242
SELECTION FLAG
220
I/O PROCESSING SECTION
210
SLOT
100
SEMICONDUCTOR RECORDING MEDIUM
110
HOST INTERFACE
120
CONTROLLER
130
DATA STORAGE SECTION
BLOCK #0
BLOCK #1
BLOCK #2
BLOCK #N-1

Fig.2

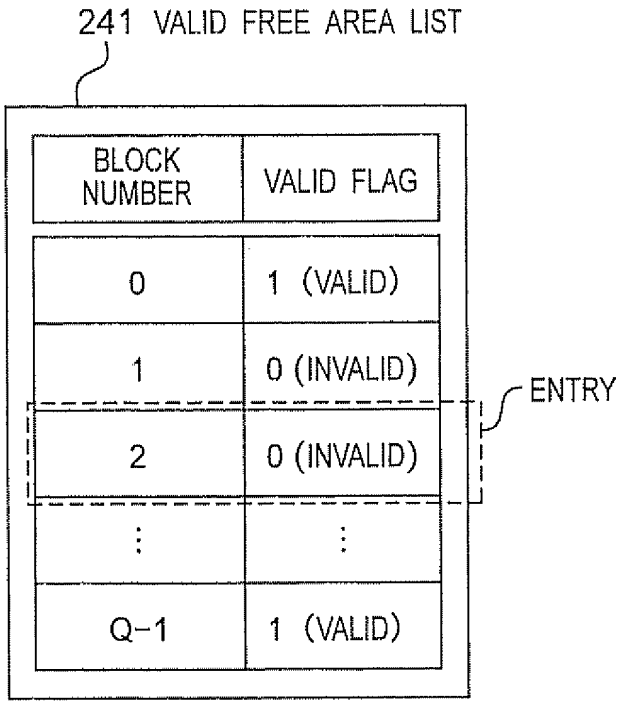
241 VALID FREE AREA LIST
BLOCK NUMBER
VALID FLAG
0
1 (VALID)
1
0 (INVALID)
2
0 (INVALID)
Q-1
1 (VALID)
ENTRY

Fig.3

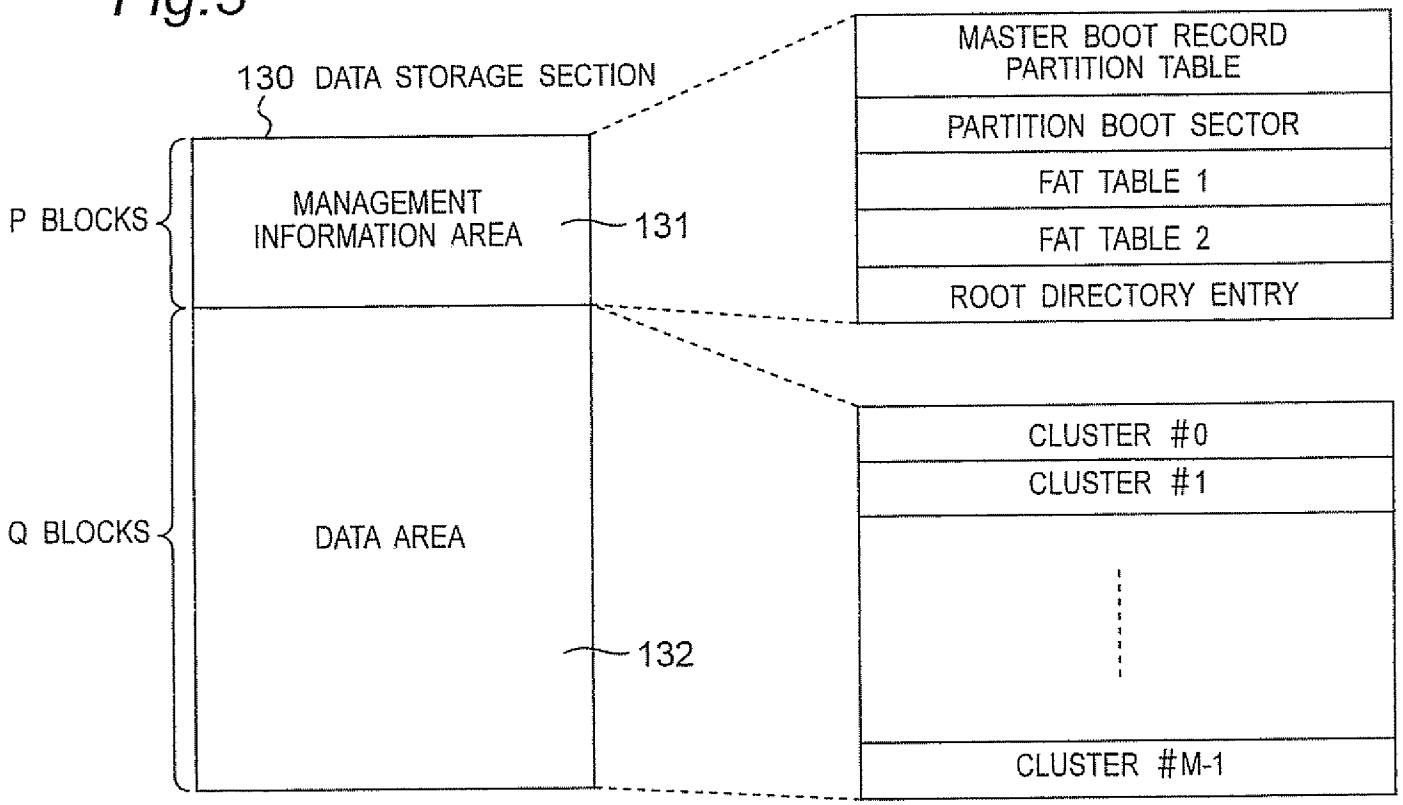
130 DATA STORAGE SECTION
P BLOCKS
Q BLOCKS
MANAGEMENT INFORMATION AREA
131
DATA AREA
132
MASTER BOOT RECORD PARTITION TABLE
PARTITION BOOT SECTOR
FAT TABLE 1
FAT TABLE 2
ROOT DIRECTORY ENTRY
CLUSTER #0
CLUSTER #1
CLUSTER #M-1

Fig.4

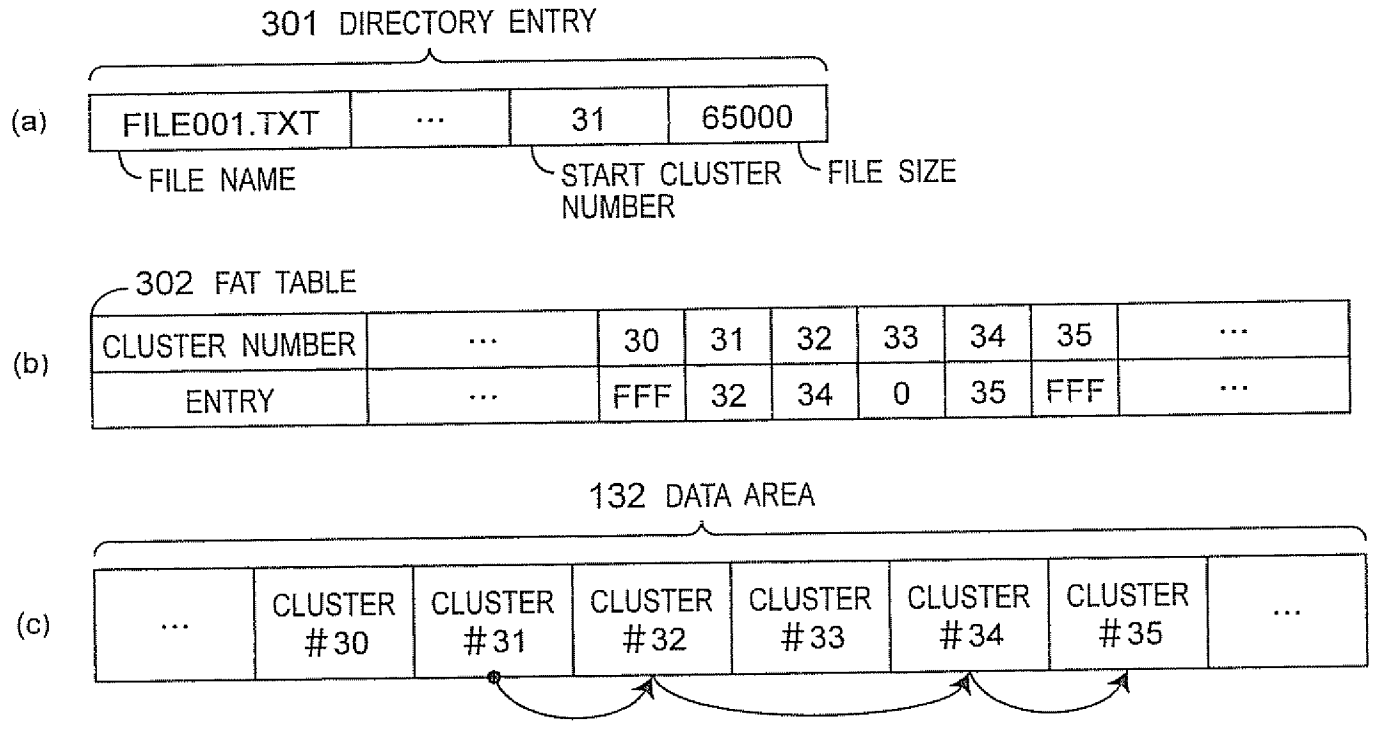
301 DIRECTORY ENTRY
(a)
FILE001.TXT
...
31
65000
FILE NAME
START CLUSTER NUMBER
FILE SIZE
302 FAT TABLE
(b)
CLUSTER NUMBER
ENTRY
30
31
32
33
34
35
FFF
132 DATA AREA
(c)
CLUSTER #30
CLUSTER #31
CLUSTER #32
CLUSTER #33
CLUSTER #34
CLUSTER #35

Fig.5

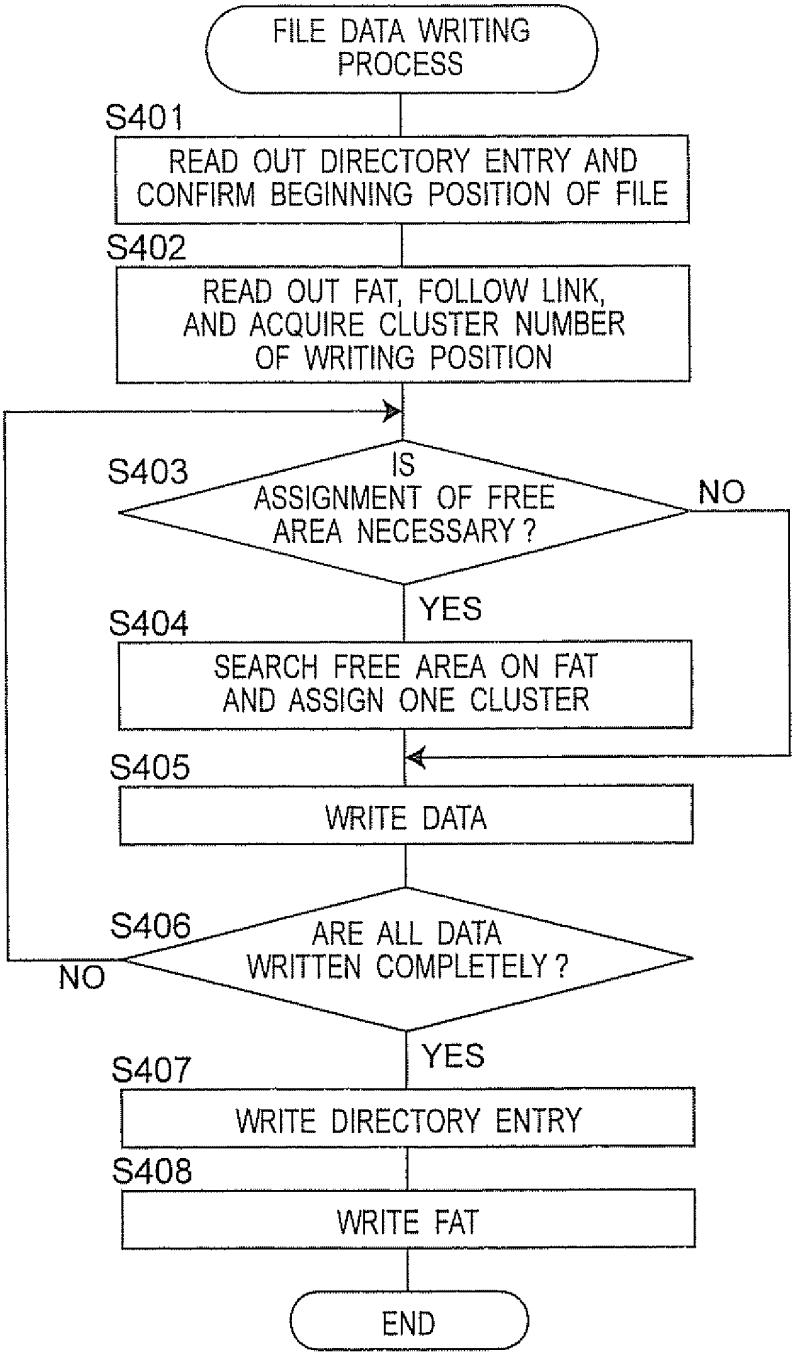
FILE DATA WRITING PROCESS
S401
READ OUT DIRECTORY ENTRY AND CONFIRM BEGINNING POSITION OF FILE
S402
READ OUT FAT, FOLLOW LINK, AND ACQUIRE CLUSTER NUMBER OF WRITING POSITION
S403
IS ASSIGNMENT OF FREE AREA NECESSARY ?
NO
YES
S404
SEARCH FREE AREA ON FAT AND ASSIGN ONE CLUSTER
S405
WRITE DATA
S406
ARE ALL DATA WRITTEN COMPLETELY ?
NO
YES
S407
WRITE DIRECTORY ENTRY
S408
WRITE FAT
END

Fig.6

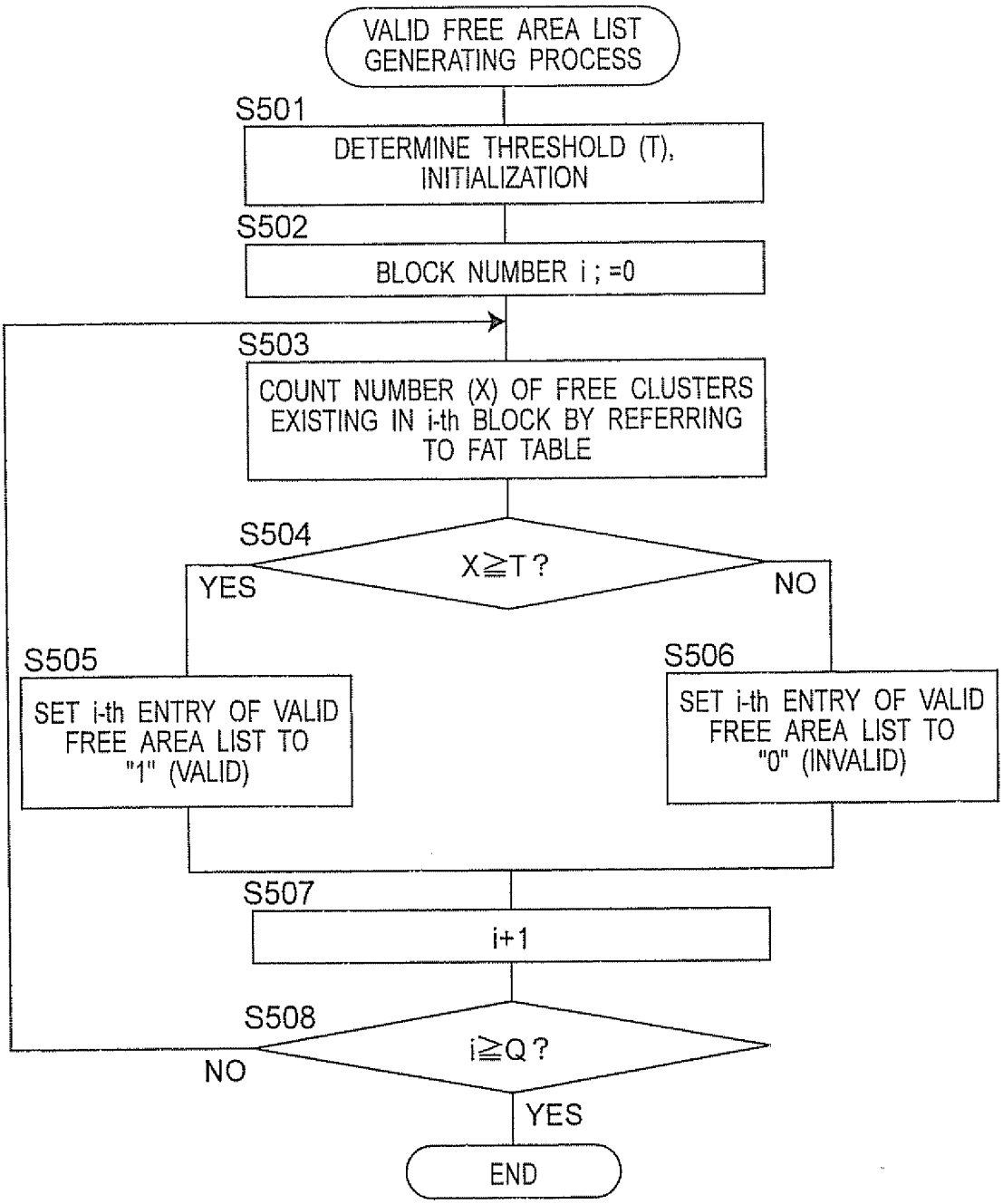
VALID FREE AREA LIST GENERATING PROCESS
S501
DETERMINE THRESHOLD (T), INITIALIZATION
S502
BLOCK NUMBER i ; =0
S503
COUNT NUMBER (X) OF FREE CLUSTERS EXISTING IN i-th BLOCK BY REFERRING TO FAT TABLE
S504
X≧T ?
YES
NO
S505
SET i-th ENTRY OF VALID FREE AREA LIST TO "1" (VALID)
S506
SET i-th ENTRY OF VALID FREE AREA LIST TO "0" (INVALID)
S507
i+1
S508
i≧Q ?
NO
YES
END

Fig.7

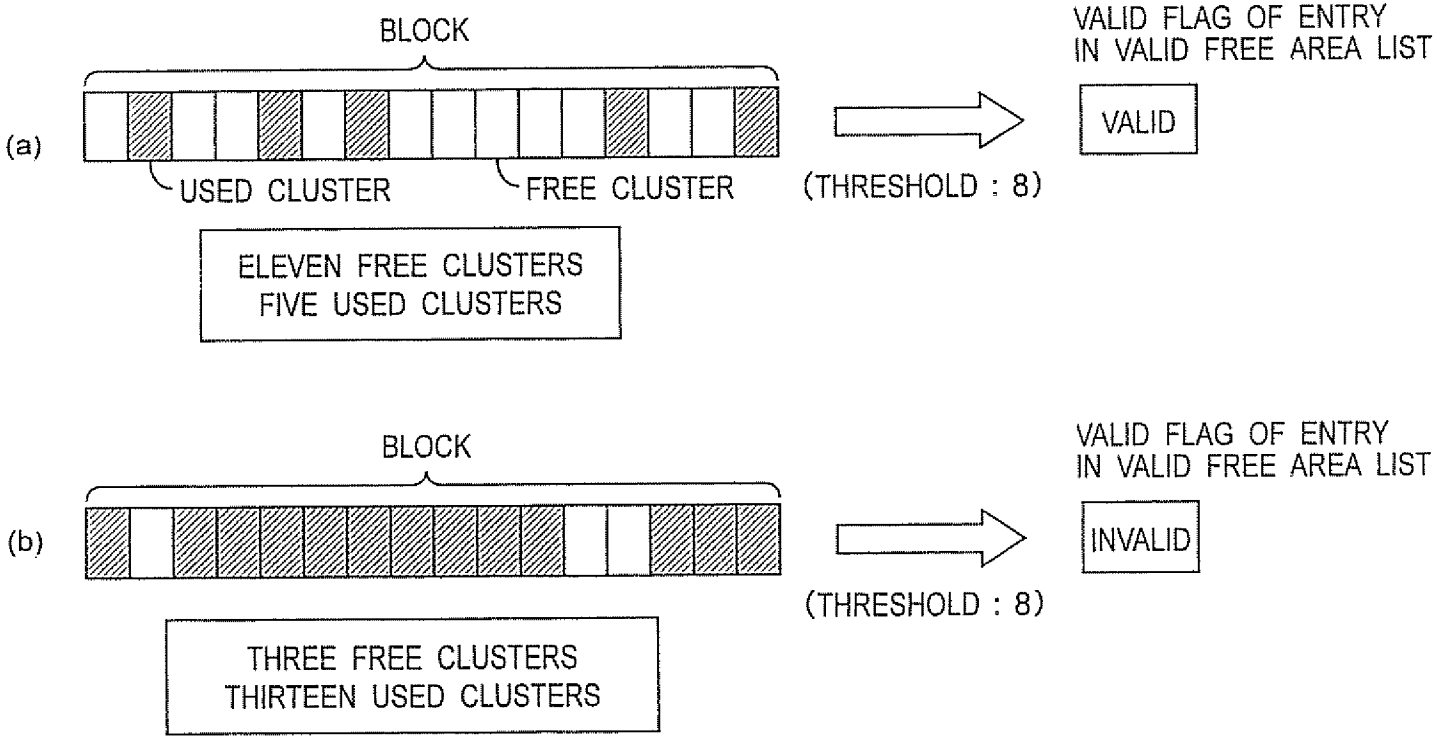
(a)
BLOCK
USED CLUSTER
FREE CLUSTER
ELEVEN FREE CLUSTERS
FIVE USED CLUSTERS
(THRESHOLD : 8)
VALID FLAG OF ENTRY
IN VALID FREE AREA LIST
VALID
(b)
BLOCK
THREE FREE CLUSTERS
THIRTEEN USED CLUSTERS
(THRESHOLD : 8)
VALID FLAG OF ENTRY
IN VALID FREE AREA LIST
INVALID

Fig.8

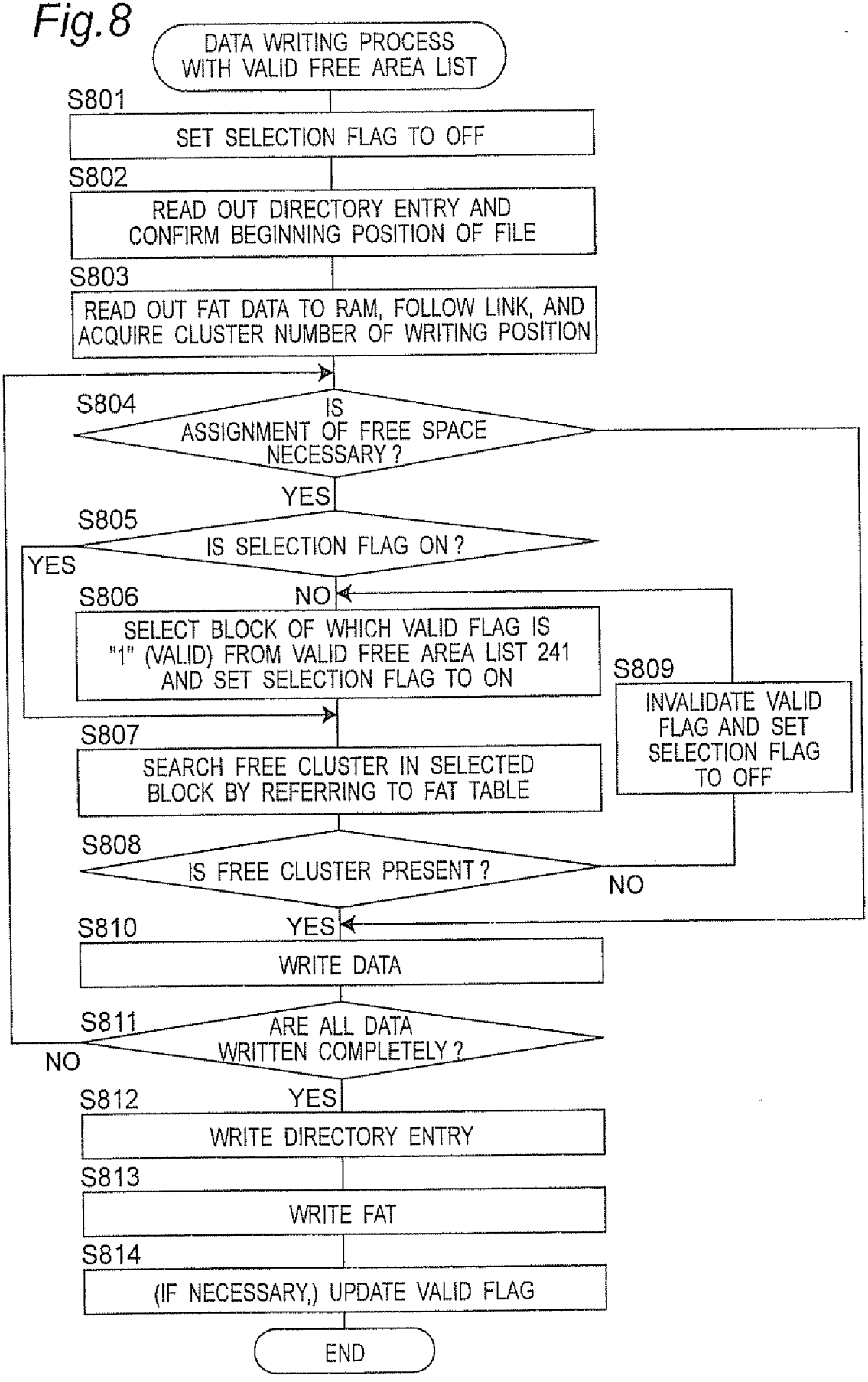
DATA WRITING PROCESS WITH VALID FREE AREA LIST
S801
SET SELECTION FLAG TO OFF
S802
READ OUT DIRECTORY ENTRY AND CONFIRM BEGINNING POSITION OF FILE
S803
READ OUT FAT DATA TO RAM, FOLLOW LINK, AND ACQUIRE CLUSTER NUMBER OF WRITING POSITION
S804
IS ASSIGNMENT OF FREE SPACE NECESSARY?
YES
S805
IS SELECTION FLAG ON?
YES
NO
S806
SELECT BLOCK OF WHICH VALID FLAG IS "1" (VALID) FROM VALID FREE AREA LIST 241 AND SET SELECTION FLAG TO ON
S809
INVALIDATE VALID FLAG AND SET SELECTION FLAG TO OFF
S807
SEARCH FREE CLUSTER IN SELECTED BLOCK BY REFERRING TO FAT TABLE
S808
IS FREE CLUSTER PRESENT?
NO
YES
S810
WRITE DATA
S811
ARE ALL DATA WRITTEN COMPLETELY?
NO
YES
S812
WRITE DIRECTORY ENTRY
S813
WRITE FAT
S814
(IF NECESSARY,) UPDATE VALID FLAG
END

Fig.9

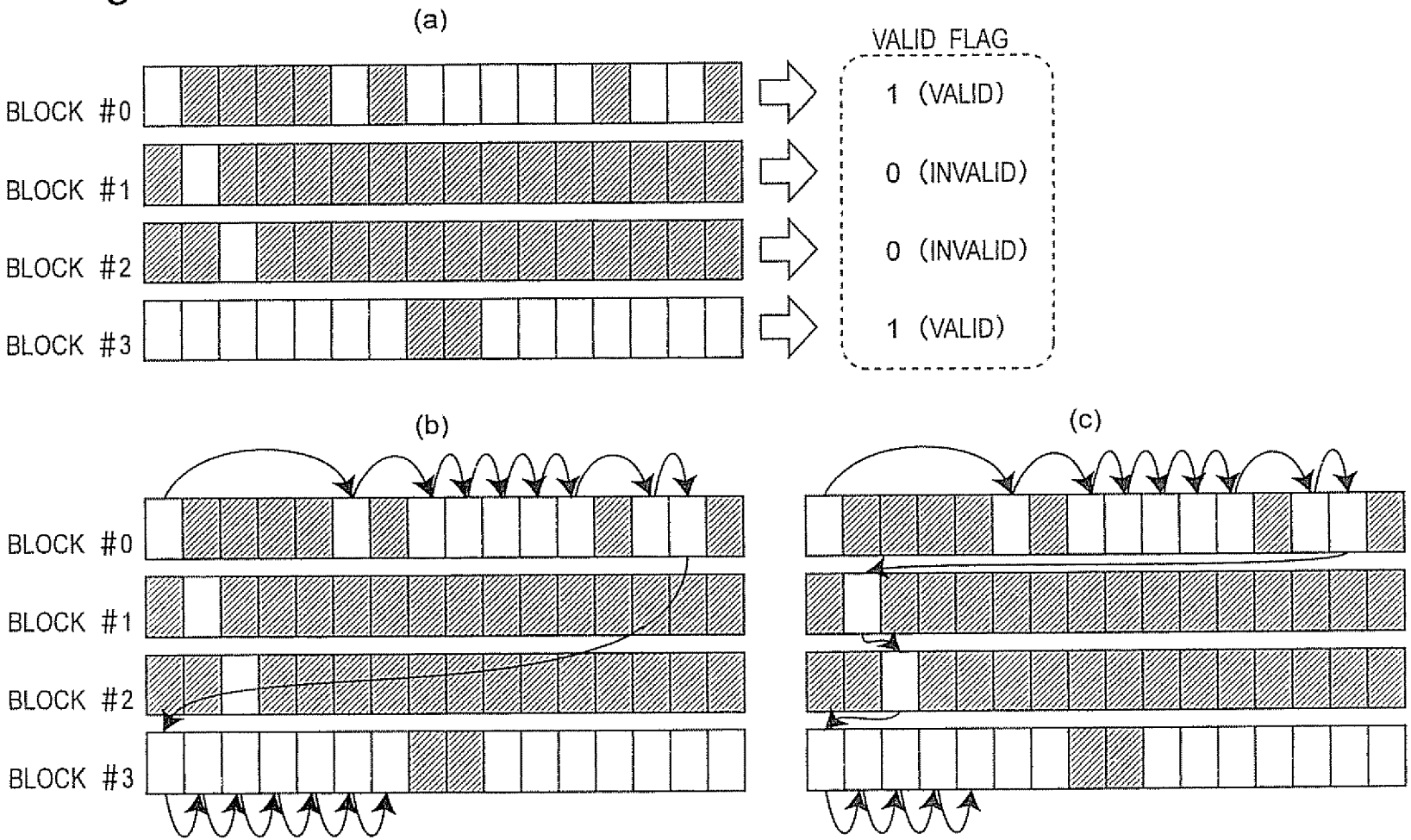
(a)
VALID FLAG
BLOCK #0
BLOCK #1
BLOCK #2
BLOCK #3
1 (VALID)
0 (INVALID)
0 (INVALID)
1 (VALID)
(b)
BLOCK #0
BLOCK #1
BLOCK #2
BLOCK #3
(c)

Fig.10

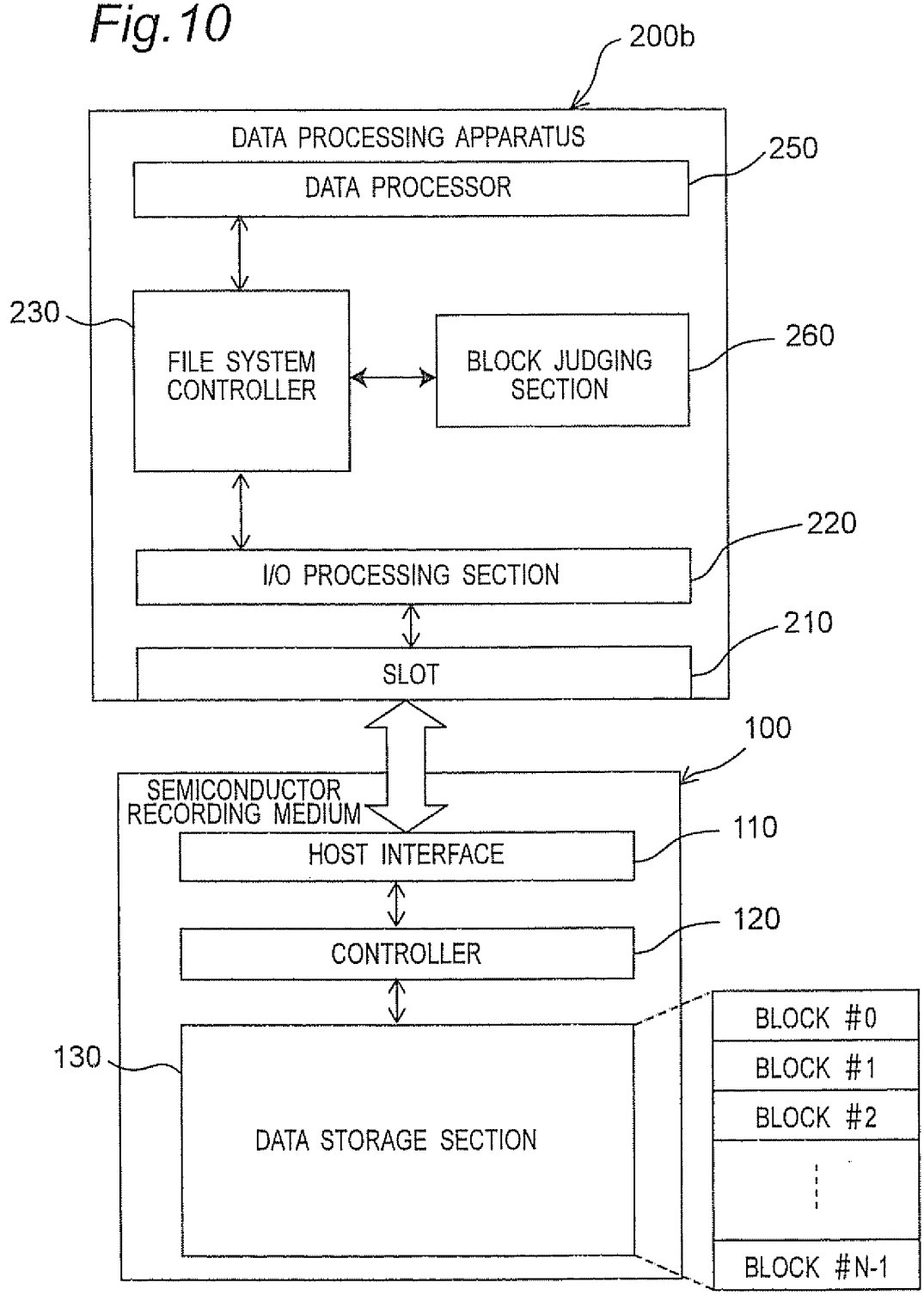
200b
DATA PROCESSING APPARATUS
250
DATA PROCESSOR
230
FILE SYSTEM CONTROLLER
BLOCK JUDGING SECTION
260
220
I/O PROCESSING SECTION
210
SLOT
100
SEMICONDUCTOR RECORDING MEDIUM
110
HOST INTERFACE
120
CONTROLLER
130
DATA STORAGE SECTION
BLOCK #0
BLOCK #1
BLOCK #2
BLOCK #N-1

Fig.11

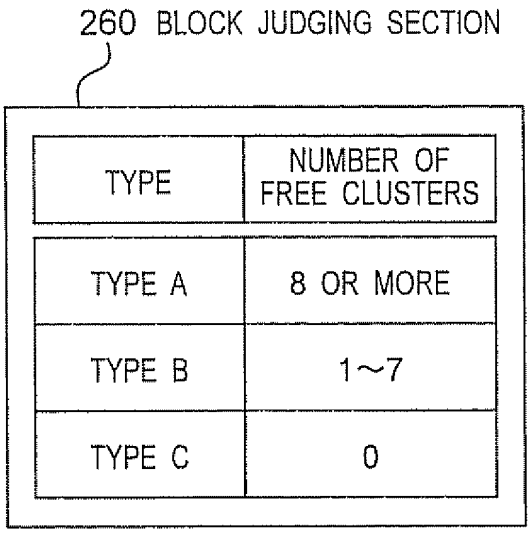
260 BLOCK JUDGING SECTION
TYPE
NUMBER OF FREE CLUSTERS
TYPE A
8 OR MORE
TYPE B
1～7
TYPE C
0

Fig.12

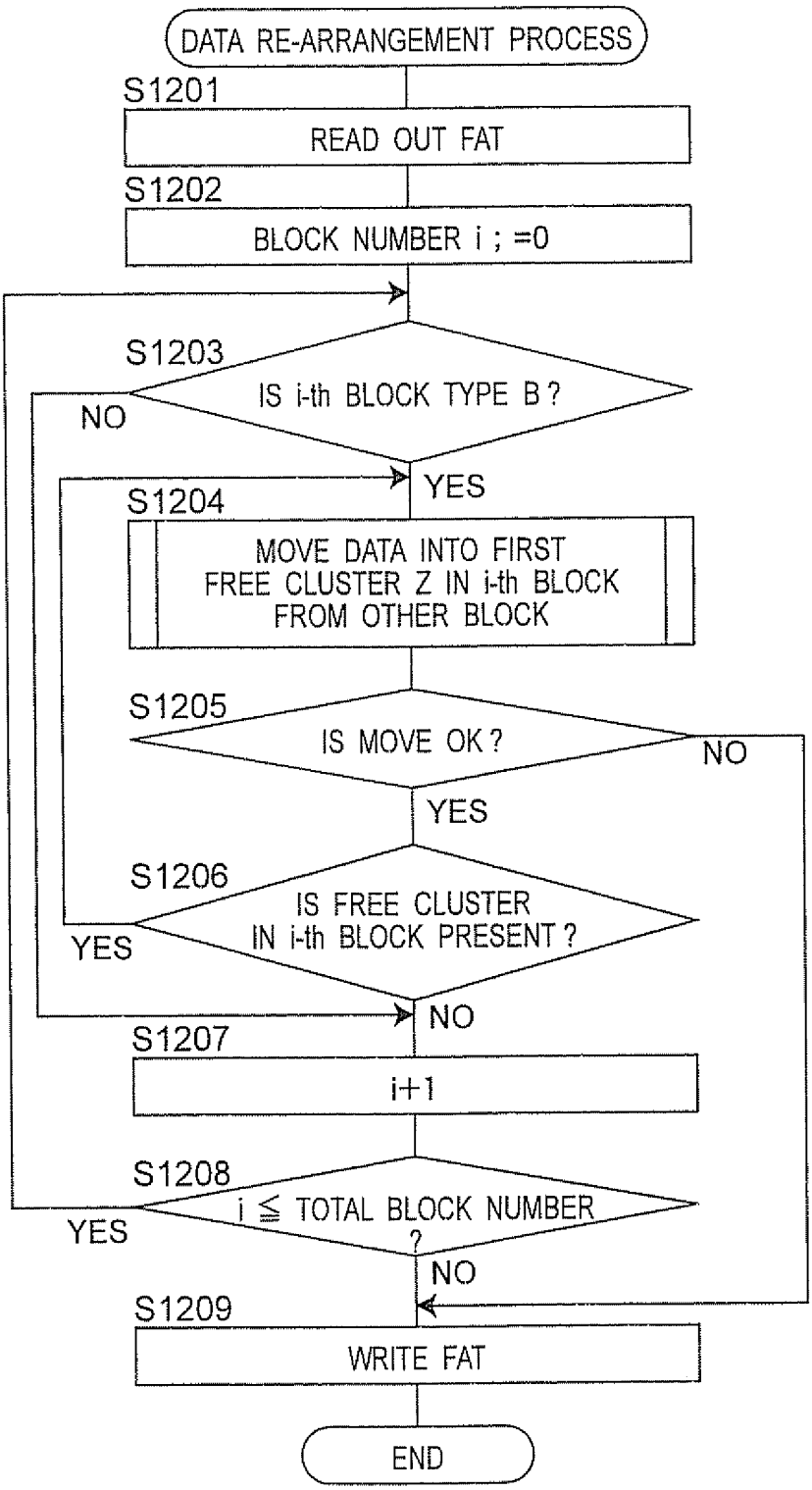
DATA RE-ARRANGEMENT PROCESS
S1201
READ OUT FAT
S1202
BLOCK NUMBER i ; =0
S1203
IS i-th BLOCK TYPE B ?
NO
YES
S1204
MOVE DATA INTO FIRST FREE CLUSTER Z IN i-th BLOCK FROM OTHER BLOCK
S1205
IS MOVE OK ?
NO
YES
S1206
IS FREE CLUSTER IN i-th BLOCK PRESENT ?
YES
NO
S1207
i+1
S1208
i ≦ TOTAL BLOCK NUMBER ?
YES
NO
S1209
WRITE FAT
END

Fig.13

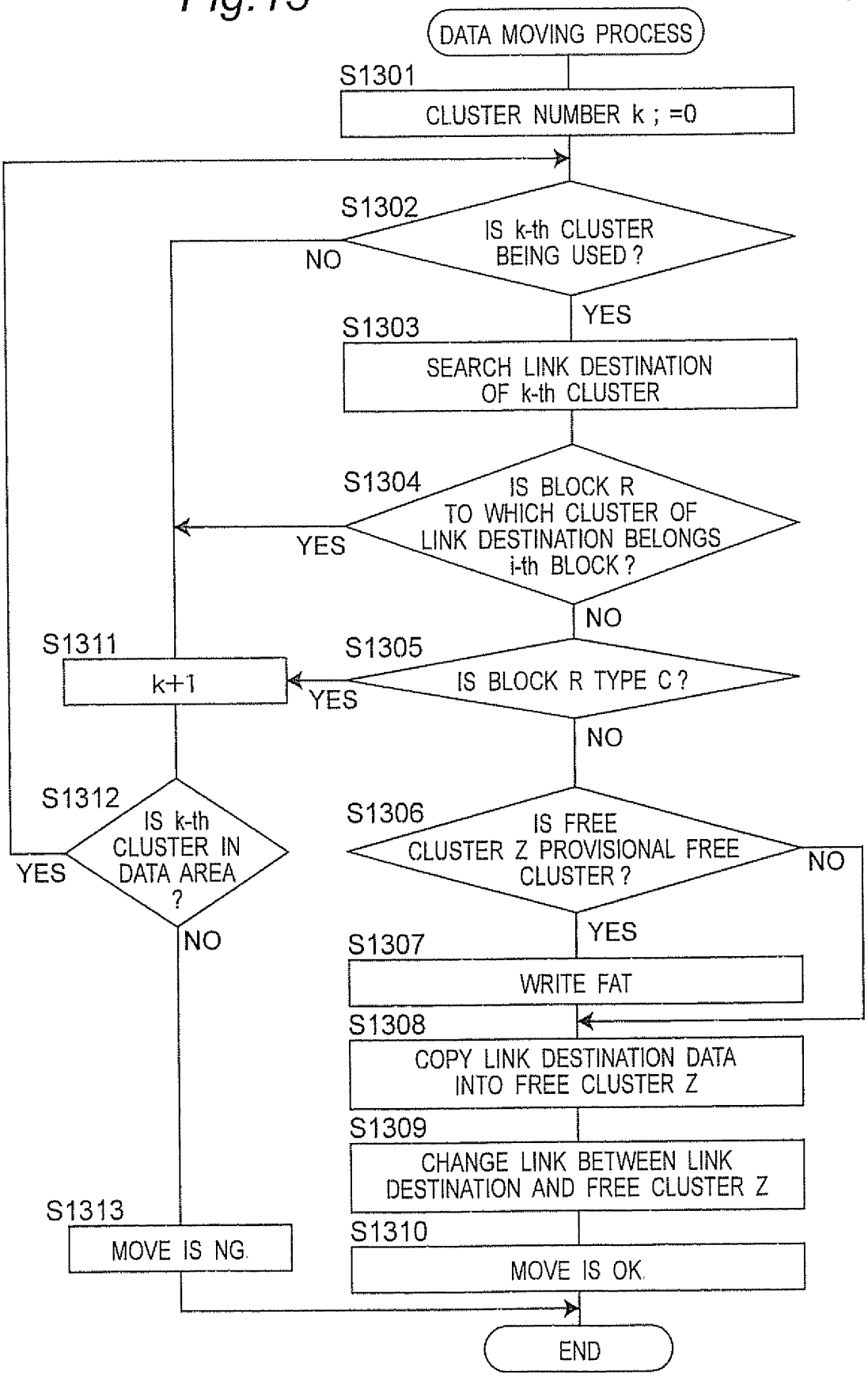
DATA MOVING PROCESS
S1301
CLUSTER NUMBER k ; =0
S1302
IS k-th CLUSTER BEING USED ?
NO
YES
S1303
SEARCH LINK DESTINATION OF k-th CLUSTER
S1304
IS BLOCK R TO WHICH CLUSTER OF LINK DESTINATION BELONGS i-th BLOCK ?
YES
NO
S1305
IS BLOCK R TYPE C ?
YES
NO
S1311
k+1
S1312
IS k-th CLUSTER IN DATA AREA ?
YES
NO
S1306
IS FREE CLUSTER Z PROVISIONAL FREE CLUSTER ?
NO
YES
S1307
WRITE FAT
S1308
COPY LINK DESTINATION DATA INTO FREE CLUSTER Z
S1309
CHANGE LINK BETWEEN LINK DESTINATION AND FREE CLUSTER Z
S1310
MOVE IS OK.
S1313
MOVE IS NG.
END

Fig.14

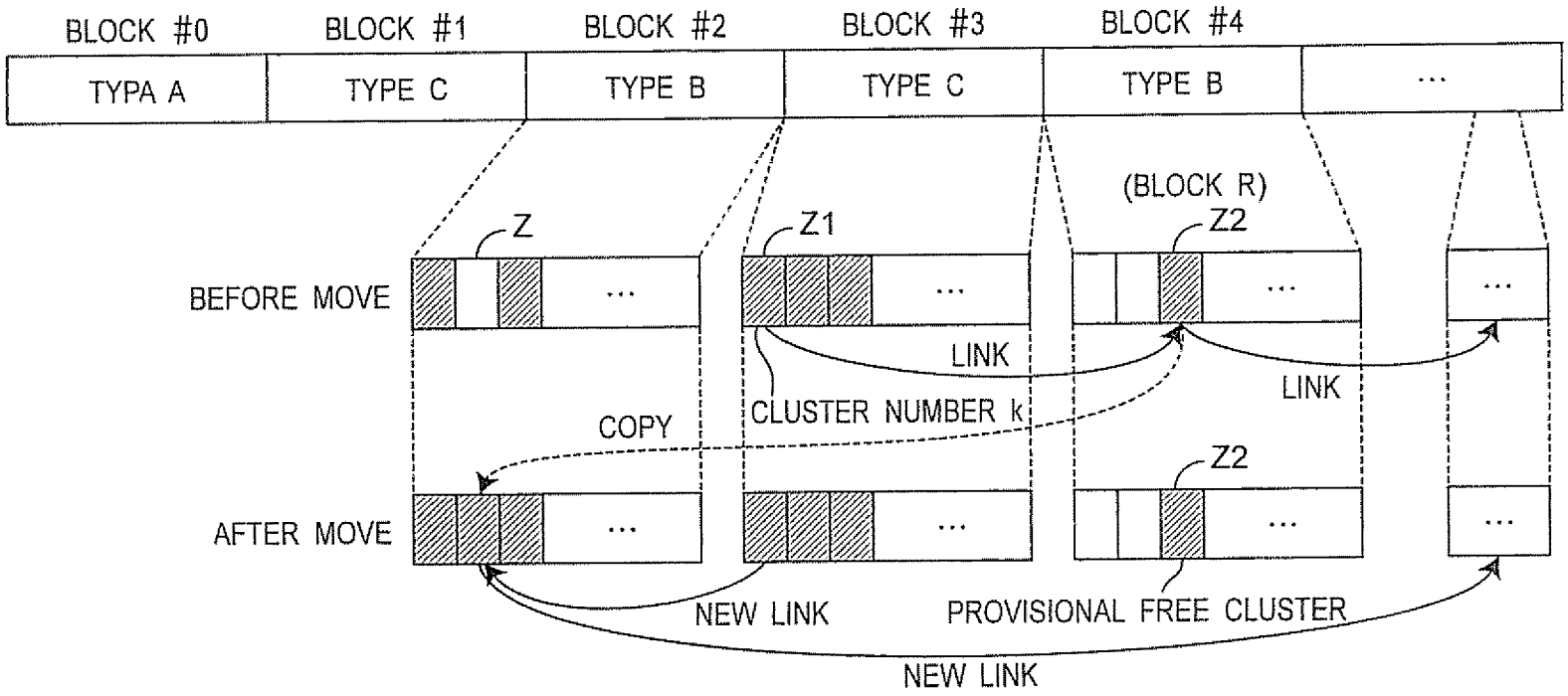
BLOCK #0
BLOCK #1
BLOCK #2
BLOCK #3
BLOCK #4
TYPA A
TYPE C
TYPE B
TYPE C
TYPE B
...
(BLOCK R)
Z
Z1
Z2
BEFORE MOVE
LINK
LINK
COPY
CLUSTER NUMBER k
Z2
AFTER MOVE
NEW LINK
PROVISIONAL FREE CLUSTER
NEW LINK

Fig.15

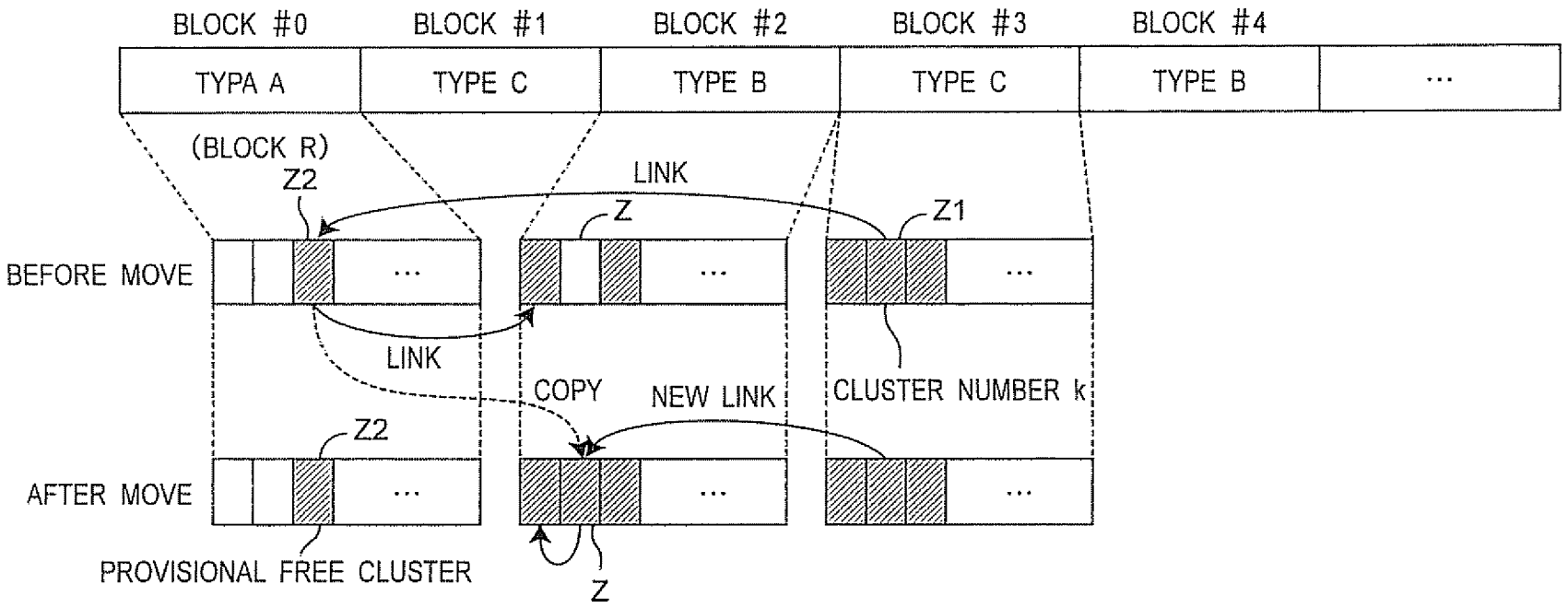
BLOCK #0
BLOCK #1
BLOCK #2
BLOCK #3
BLOCK #4
TYPA A
TYPE C
TYPE B
TYPE C
TYPE B
...
(BLOCK R)
Z2
LINK
Z
Z1
BEFORE MOVE
LINK
COPY
NEW LINK
CLUSTER NUMBER k
Z2
AFTER MOVE
PROVISIONAL FREE CLUSTER
Z

Fig.16A

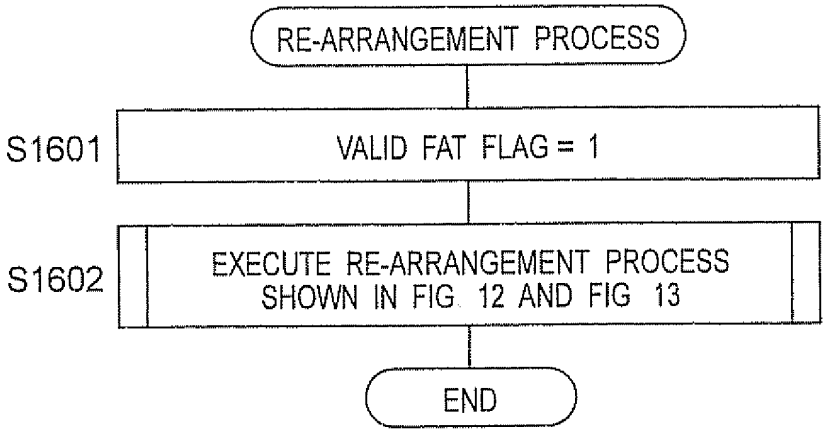
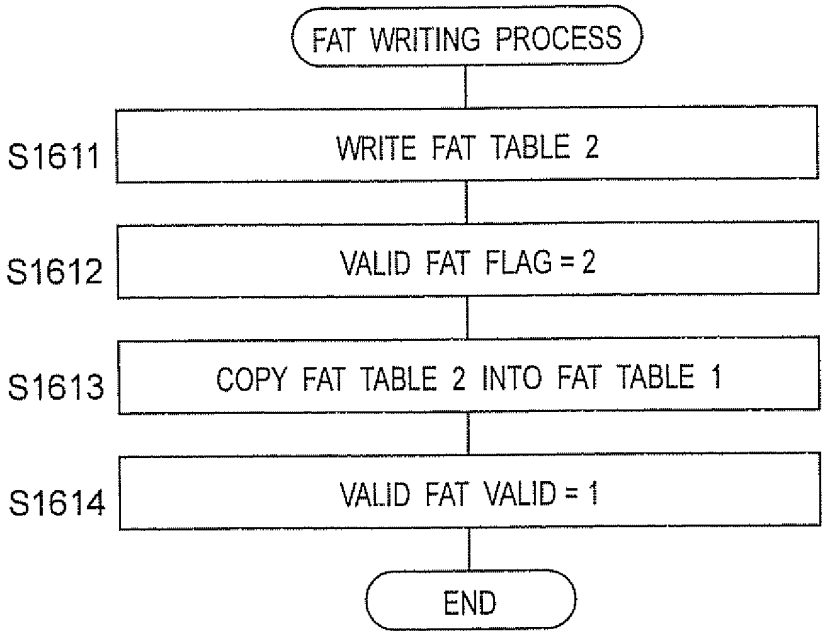
Fig.16B